Similarly, Gulick fails to teach or suggest that non-preferred applications are automatically disabled after preferred applications are enabled as required by claim 1. To this end, Gulick teaches that when an application is attempted to be initiated and insufficient resources are available to perform the application, a user may, in at least some cases, disable an application currently executing to free resources for execution of the new application. Here, Gulick fails to teach or suggest that the application to be disabled is a non-preferred application and thus fails to teach the step of identifying a non-preferred application. Instead, Gulick simply teaches that a currently executing application is identified and disabled without providing any comments regarding how the application to be disabled should be selected (i.e., Gulick does not contemplate that some applications will be non-preferred). In fact, Applicant notes that usually applications that are executed when a workstation user accesses a workstation are preferred applications – the general meaning behind the qualifier "preferred" is that an associated user uses or causes applications to execute when the user is using the workstation. By teaching that an executing application is disabled, while Gulick may stop short of teaching away from identifying non-preferred applications that are booted, Gulick clearly demonstrates that Gulick did not contemplate distinguishing between different booted applications (i.e., between preferred and non-preferred applications) for disablement.

In addition, claim 1 requires identifying all applications that are currently booted that are non-preferred, not just one or the number of currently executing applications that need to be disabled to allow another application to be executed. Gulick, however, teaches only identifying a number of executing applications that need to be disabled in order to free up resources required to execute another application. Here, by only disabling the number of currently executing applications that need to be disabled in order for another application to be executed, the processing power of a system is always pushed to its maximum which is a much different result than disabling all non-preferred applications. For instance, in one exemplary case where a physician only has one preferred application and all other applications are non-preferred, according to claim 1, only the one preferred application would remain booted after the claim 1 process is completed and the other applications would all be disabled. In contrast, according to Gulick, the end result may be that nine of the non-preferred applications remain booted despite a preference to have those other nine applications disabled so that the one preferred application could be executed more quickly.

For at least the reasons described above, Applicant believes that claim 1 and claims that depend therefrom are patently distinct over the combination of the references cited in the Office Action and Applicant requests that the rejection of those claims be withdrawn.

With respect to each of claims 10, 18 and 20, each of those claims includes limitations similar to the limitations of claim 1 that require identification of non-preferred applications and disablement of non-preferred applications. Thus, for the same reasons that Applicant believes claim 1 is distinct over the cited references, Applicant believes each of claims 10, 18 and 20 is also patently distinct and requests that the rejections of those claims and the claims that pend therefrom be withdrawn.

20-22. With respect to claim 8, as recognized by the Office Action, none of the Microsoft reference, Watts or Gulick teaches or suggests identifying critical applications associated with preferred applications and adding the critical applications to the preferred list of applications for a user. Walsh fails to teach what the other references lack.

Turning to the specific sections of Walsh cited in the Office Action, Walsh teaches that when any program is launched (i.e., execution is commenced), various program files have to be executed in order to perform the launch process. To this end, Walsh teaches that a load sequence list is pre-stored that specifies the order in which the program files have to be executed to perform the launch. Walsh also teaches that in at least some cases the program files to perform a launch will not be arranged in a sequential order and therefore a processor performing a launch process will be slowed up as the processor will have to perform additional addressing steps in order to identify the locations of the program files during the launch process. To overcome this problem, Walsh teaches that prior to performing a launch process, launch related files can be identified and restored in a contiguous order so that the addressing steps can be avoided. During the reordering process, Walsh teaches that the load sequence list that specifies the order of file execution during a launch process is accessed and the launch files are reordered and restored accordingly.

Walsh's reordering and restoring of launch required files has nothing to do with identifying applications that are critical to preferred applications associated with a specific system user. In addition, Walsh's reordering and restoring has nothing to do with identifying critical applications associated with some other application and supplementing the critical applications list as a function of identified critical applications. Instead, Walsh's load sequence list pre-exists and is never supplemented (i.e., other launch files are never added to the list). Thus, whatever Walsh teaches, clearly Walsh fails to teach or suggest any step that is associated with a user preferred application as required by claim 8. In addition, whatever Walsh teaches, Walsh clearly fails to teach or suggest any step that adds or in any other way supplements a list of applications of any type – again, Walsh's load sequence list pre-exists unlike the user's preferred list of applications in claim 8 which is supplemented.

For at least the reasons described above Applicant believes claim 8 is independently patently distinct over the cited references and requests that the rejection of claim 8 be withdrawn.

Each of claims 16 and 19 includes limitations similar to the limitations of claim 8 and for similar reasons Applicant believes that each of claims 16 and 19 is patently distinct over the cited references and requests that the rejections of those claims be withdrawn.

In view of the above remarks, Applicant believes claims 1, 3-10 and 12-20 of the present application recite patentable subject matter and allowance of the same is requested. No fee in addition to the fees already authorized in this and accompanying documentation is believed to be required to enter this amendment, however, if an additional fee is required, please charge Deposit Account No. 07-0845 in the amount of the fee.

Respectfully submitted,

Phani Kumar Bidarahalli

Date: 5-16-05

By: Michael A. Jaskolski
Reg. No. 37,551
Attorney for Applicant
QUARLES & BRADY, LLP
411 East Wisconsin Avenue
Milwaukee, WI. 53202-4497
(414) 277-5711